

Appendix A

Agent Descriptions

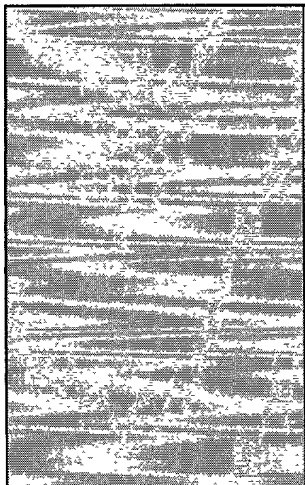
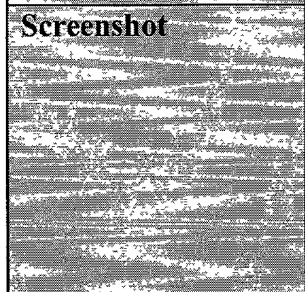


ATG Dynamo

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	UAP
Typically Updated	10 minutes
Collection Method	SNMP
Statistics Returned	d3System <ul style="list-style-type: none"> • sysServerName • sysStatus • sysUpTime • sysTotalMem • sysFreeMem • sysNumInfoMsgs • sysNumWarningMsgs • sysNumErrorMsgs d3LoadManagement <ul style="list-style-type: none"> • lmIsManager • lmManagerIndex • lmIsPrimaryManager • lmServicingCMs d3SessionTracking <ul style="list-style-type: none"> • stCreatedSessionCnt • stValidSessionCnt • stRestoredSessionCnt • stDictionaryServerStatus d3DRPSTServer <ul style="list-style-type: none"> • drpTotalReqsServed • drpTotalReqTime • drpAvgReqTime • drpNewSessions d3DBConnPooling <ul style="list-style-type: none"> • dbPoolID • dbMinConn

	<ul style="list-style-type: none"> • dbMaxConn • dbMaxFreeConn • dbBlocking • dbConnOut • dbFreeResources • dbTotalResources
Bugs	None known
Implementation Details	Collects various stats from the Dynamo server via SNMP. Configured through the "dynagent.cfg" file. This script could easily be used to create other SNMP agent; the configuration file style makes it very versatility.

CommerceTrends

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	TSN
Typically Updated	Daily
Collection Method	Parsing text file output from CommerceTrends
Statistics Returned	<p>Daily Stats:</p> <ul style="list-style-type: none"> • Page Views • Page Views % of Month • Dwell Time • First-time Visits • Unique Visitors • Visit Minutes • Visits <p>Monthly Stats:</p> <ul style="list-style-type: none"> • Page Views • Visits • Unique Users • Users Visited Multi • Users Visited Once • Return Visits • First-time Visits • No-cookie Visits
Bugs	None known
Implementation Details	<p>CommerceTrends uses Perl scripts to generate all of its reports.</p> <p>/usr/local/commercetrends/wt_script/report.pl was modified to output statistics to a flat text file as the reports</p>

	<p>were generated. The text files live at: vp3:/u05/commercetrends/xordata/incoming.</p> <p>CommerceTrends is configured to run the script /info/dashboard/bin/ct_postprocess <i>sitenum</i> when it finishes processing the logs. This script adds the information in the text file to a DBM file for the specified site. Another script, ct_agent, is called to generate the XML file that's picked up by the reaper.</p> <p>Nearly any statistic that shows up in a report can be imported into VisionPort. It just takes some changes to each of the scripts mentioned above.</p>
	<p>8 Days Ago </p> <p>1,651,292 pv</p> <p>As of May 22 2001, 12:00am:  1 d</p> <p>337,678 visits</p> <p>265,198 unique visitors</p> <p>44,008 new visitors</p> <p>5:03 dwell time</p>

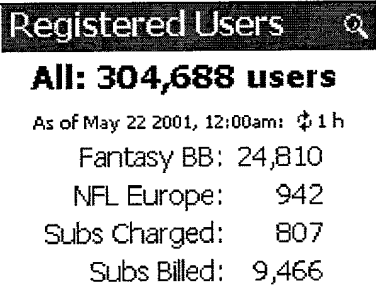
Foglight

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	XOR
Typically Updated	15 minutes
Collection Method	Text Scraping
Statistics Returned	Machine Load and CPU usage
Bugs	None known
Implementation Details	Takes the Foglight site identifier in on the PATH_INFO. For example: NT_system@web-nt8.xor.com. Based on the type of system (determined from looking at the site identifier for "NT", "vision2" (bsdi), or "Sun", it executes a foglight command that outputs stats for that system for the past hour. Other statistics can be gathered. See the script for pointers to the foglight commands used to determine the statistics that are available.

HTTP/Text Scraping Example

Language	Java
Version	1.0
Reusability Level	High
Customers Using	None
Typically Updated	N/A
Collection Method	HTTP/Text scraping
Statistics Returned	Stock ticker
Bugs	None known
Implementation Details	Java version of the Perl-based quote fetcher for Yahoo stock ticker.

Informix DB Data Extraction


Language	Perl
Version	1.0
Reusability Level	Low; TSN-specific
Customers Using	TSN
Typically Updated	10 minutes
Collection Method	SQL Queries
Statistics Returned	Registration information from the TSN user database
Bugs	None known
Implementation Details	TSN has lots of user data stored in an Informix database that lives on tsn-db1. Scripts on tsn-db2 connect to the DB, extract relevant information with SQL queries, and format it into XML.
Screenshot	 <p>Registered Users</p> <p>All: 304,688 users</p> <p>As of May 22 2001, 12:00am: 1 h</p> <p>Fantasy BB: 24,810</p> <p>NFL Europe: 942</p> <p>Subs Charged: 807</p> <p>Subs Billed: 9,466</p>

PingTier1

Language	Perl
Version	1.0
Reusability Level	XOR-specific
Customers Using	MCC
Typically Updated	1 day
Collection Method	Text file scraping
Statistics Returned	Average and Maximum ping time to XOR's tier-1 providers over the past 31 days.
Bugs	None known
Implementation Details	Some shell scripts ping each of XOR's tier-1 provider connections every 5 minutes and store the results for 31 days. The agent looks at the files and averages the maximum and average ping times.

RealMedia OpenAdstream

Language	Perl
Version	1.0
Reusability Level	Low, unless another client uses OpenAdstream

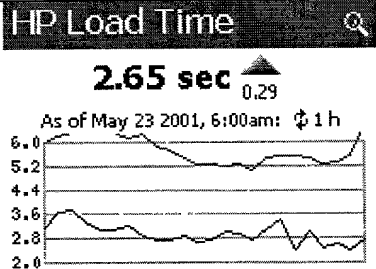
Customers Using	TSN
Typically Updated	10 minutes
Collection Method	Log parsing
Statistics Returned	Number of ads served per ad position.
Bugs	None known
Implementation Details	Constantly reads the OAS log files on each machine and counts the number of as served. The agent reads a DBM file (the same one as the near-real time log parser.
Screenshot	 <p>Today's Ads</p> <p>Total: 2,402,605</p> <p>(171.34/s)</p> <p>As of May 23 2001, 9:07am: 10 m</p> <p>Top: 1,519,945 (108.1/s)</p> <p>TopRight: 249,295 (18.9/s)</p> <p>BottomLeft: 130,503 (9.3/s)</p>

Remedy

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	XOR
Typically Updated	Hourly
Collection Method	SQL Queries against Remedy database
Statistics Returned	Requests, Pending, Resolved, Min/Max/Avg TTR for each priority. Service Communications (email/phone). List of all incidents Started, Resolved, or Pending in a given time period.
Bugs	The Service Communications piece may not be accurate, as it's unclear how tickets that are opened automatically from HPOV are logged as "communicating".
Implementation Details	Connects via the following: 'dsn' => 'dbi:Oracle:host=evolve.xor.com;sid=ARSPR1', 'username' => 'vision_report', 'password' => 'vision_report'

ServiceMetrics

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	TSN

Typically Updated	Hourly
Collection Method	Incoming email parsed
Statistics Returned	Page load time history for the past 24 hours Service Metrics' Internet Index and eCommerce Index
Bugs	Sometimes one data item has an additional field (data that it has collected so far this hour). If this data field exists, it should be deleted.
Implementation Details	ServiceMetrics was configured to email a report to servicemetrics@xor.com . This email address pipes the output to a script that sends the data over to dev-linux for processing. On dev-linux, a script run from cron every few minutes looks for incoming files and delivers them to the correct programs for processing.
Screenshot	 <p>The screenshot shows a graph titled 'HP Load Time'. At the top, it displays '2.65 sec' with a small triangle icon and '0.29' below it. Below this, it says 'As of May 23 2001, 6:00am: 1 h'. The graph itself has a y-axis ranging from 2.0 to 6.0 in increments of 0.8. It shows a line graph with two distinct lines, one generally higher than the other, representing load times over a period of time.</p>

SQL Query Skeleton

Language	Java
Version	1.1
Reusability Level	High
Customers Using	None
Typically Updated	N/A
Collection Method	SQL Queries
Statistics Returned	Columns/rows requested through query
Bugs	None known
Implementation Details	<p>First-pass Java SQL query agent. It returns generic information, based on the SQL query specified to the agent. Arguments:</p> <ul style="list-style-type: none"> • URL: string used to access the database • username: database username • password: database user password • labeledColumns: Boolean: whether or not to label columns in the XML output. • labeledNumeric: Boolean: if the labels should be numeric or strings; if strings, they're the SQL column names. • labeledZeroBased: Boolean: if the labels are 0- or 1-based. Even if labeledNumeric is false, this

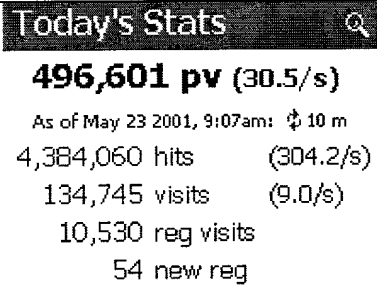
	<p>variable is still used to determine how the returned "label" item's names are output as well as the returned rows' names.</p> <ul style="list-style-type: none"> • query: SQL query to execute
--	--

System Statistics

Language	Perl
Version	1.0
Reusability Level	High, but only on same operating system. Currently implemented on BSDI and Linux. Easy to port to another O.S.
Customers Using	TSN
Typically Updated	10 minutes
Collection Method	System Calls
Statistics Returned	1, 5, and 15 minute load time Uptime of the machine in seconds ("elapsed" data type) Approximate machine CPU utilization
Bugs	None known
Implementation Details	<p>Uses a system call to "uptime" to gather 1, 5, and 15 minute load times and the uptime of the machine.</p> <p>To gather information about CPU utilization, it uses:</p> <ul style="list-style-type: none"> • BSDI: /usr/sbin/iostat -c 2 10 • Linux: Looks at /proc/stat in 10 second intervals
Screenshot	<p>Avg System Load</p> <p>2.92 ▲ 0.12</p> <p>As of May 23 2001, 9:07am: ↕ 10 m</p>

Web Log Statistics (XOR): Near-realtime

Language	Perl
Version	1.0
Reusability Level	High on systems that use standard web logging. Would need to be modified to be used on NT.
Customers Using	TSN
Typically Updated	10 minutes
Collection Method	Parsing web logs on-the-fly
Statistics Returned	For today since midnight: Hits, page views, visits, page

	views per visit, page views per content area
Bugs	None known
Implementation Details	A daemon runs on individual machines and parses the web log files on-the-fly. Every 10 minutes the daemon outputs the current statistics to a DBM file (/info/httpd/statsum/YYMM/sitename/dashboard). The agent reads this DBM file to generate its XML.
Screenshot	 <p>Today's Stats 🔍</p> <p>496,601 pv (30.5/s)</p> <p>As of May 23 2001, 9:07am: ⌚ 10 m</p> <p>4,384,060 hits (304.2/s)</p> <p>134,745 visits (9.0/s)</p> <p>10,530 reg visits</p> <p>54 new reg</p>

Web Log Statistics (XOR): Historical

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	TSN
Typically Updated	Daily
Collection Method	Extract statistics from DBM file generated from XOR log parser
Statistics Returned	<p>Hits, page views, visits, page views per visit, page views per content area for:</p> <ul style="list-style-type: none"> • Yesterday • 8 days ago • 7 day average • 28 day average <p>A 28 day running history is also available for "visitors".</p>
Bugs	None known
Implementation Details	A program on the log parsing machines (web-log and tslogs) analyzes log files nightly. When it's done with a day, it outputs stats for the day into a DBM file (/info/httpd/statsum/YYMM/sitename/misc). The agent reads this DBM file to generate its XML.

Web Log Statistics (XOR): Historical – Last Month Only

Language	Perl
Version	1.0

Reusability Level	High
Customers Using	MCC
Typically Updated	Daily
Collection Method	Extracts statistics from past monthly reports by text scraping.
Statistics Returned	Hits, page views, visits, page views per visitor, megabytes transferred area for, this month, last month, and a year ago last month.
Bugs	None known
Implementation Details	Scrapes files on web-log.xor.com from /info/logs/old-logs/YYYY/Mon/www/sitename. Is able to deal with web log reports that have been gzipped as well. The script gets the sitename from the PATH_INFO environment variable. It's called like: http://web-log.xor.com/cgi-bin/agents/weblog_stats_monthly_history/www.sitename.com

WebPosition Gold

Language	Perl
Version	1.0
Reusability Level	High
Customers Using	None
Typically Updated	Weekly
Collection Method	HTTP calls, scraping the returned tab-delimited text
Statistics Returned	Number of keywords searched for, total number of matches, average position, and for each search engine: <ul style="list-style-type: none"> • Number of matches • Positions for each match • Average position
Bugs	None known
Implementation Details	WebPosition gathers information about how a site is ranked on search engines. It looks like the only way to get information out of it is through scraping text files that it generates. It only runs on NT. WebPosition is running on web-nt7. URLs to get the data is in the format: web-nt7.xor.com/reports/www.sitename.com.txt

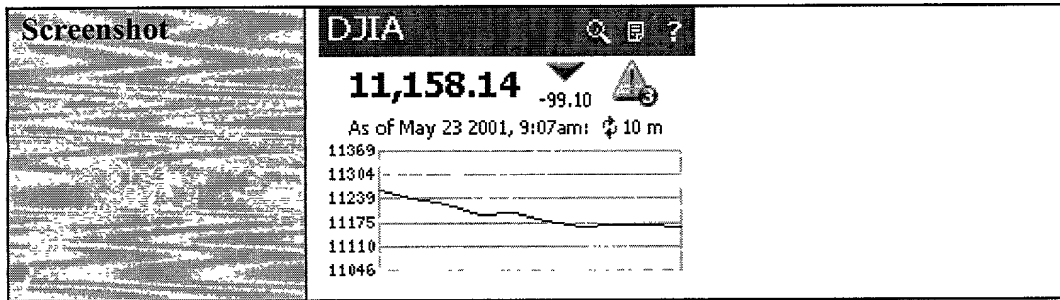
Screenshot	Search Engines
	Matches: 91 (34.21%) As of May 23 2001, 8:43am: 7 d In Top 10: 46 (17.29%) Searches: 266 (19 sites) Avg Position: 12.7 URLs Found: 44

Yahoo News

Language	Perl
Version	1.0
Reusability Level	High, but this probably isn't legal
Customers Using	None
Typically Updated	Hourly
Collection Method	HTTP calls, scraping the returned HTML
Statistics Returned	Current news for a variety of topics (World, Tech, Finance, etc)
Bugs	None known
Implementation Details	Calls URLs for different types of news. Examples: Headlines: http://dailynews.yahoo.com/headlines/ts/ Business: http://dailynews.yahoo.com/headlines/bs/ It the scrapes the top news stories out of the HTML.

Yahoo Stocks

Language	Perl
Version	1.1
Reusability Level	High, but this probably isn't legal
Customers Using	TSN
Typically Updated	10 minutes
Collection Method	HTTP calls, scraping the returned HTML
Statistics Returned	Today's history of the following indexes: NYSE, NASDAQ, DOW, and S&P 500. Can easily be customized to get stats for specific stocks
Bugs	The first quote of the day isn't recorded in the reaper repository. This is likely a bug with the reaper when it receives a "clear history" instruction.
Implementation Details	Call this URL to get stock information: http://quote.yahoo.com/quotes?SYMBOLS=^DJI,^IXIC,^SPX Note that other symbols can be added to the end to capture other quotes.



Agents Under Development

HP Openview

Language	Perl
Priority	High
Reusability Level	High
Customers	MCC, dashboards for clients
Typically Updated	10 minutes
Description	HPOV monitors machines/equipment from a networking level. This mostly determines if a machine or piece of networking hardware is available or not.

Proposed Agents

Keynote

Likely Language	Perl
Reusability Level	High
Customers	Qwest
Typically Updated	Hourly
Description	Keynote is a site monitoring service, similar to ServiceMetrics.

cPulse

Description	XOR's clients use cPulse to deliver customer satisfaction surveys.
--------------------	--

MRTG

Likely Language	Perl
Priority	Probably being scrapped in favor of OpenView stats
Reusability Level	High
Customers	
Typically Updated	10 minutes
Collection Method	Text file scraping
Statistics Returned	Current, average, 95 th percentile of bandwidth
Description	MRTG gathers live bandwidth statistics via the hosting infrastructure's switches.

Appendix B

Dashboard Data Types

dataitem

Description	The base class from which nearly all other data types inherit their default behavior.
Notes	Abstract class
Attributes	
name	The name of this particular piece of information. The name can only contain the following characters: A-Za-z0-9_.- (no spaces or punctuation!)
description	A description for this piece of information
type	The data type of this particular piece of information
timestamp	The time the current value was sampled
XML	None. The raw <i>dataitem</i> type should not be used directly.

image (isa dataitem)

Description	A pointer to an image URL
Notes	This has yet to be implemented in the controller
Attributes	
image	The URL of an image to display
XML	<code><image name="status_image" image="http://visionport.xor.com/images/ok.gif"/></code>

text (isa dataitem)

Description	Holds plain text. The plain text may contain newlines and leading spaces for formatting purposes.
Notes	The formatting may be significant and should be respected by the GUI if possible. The Controller may reformat the text for display on a given device.
Attributes	
text	The text to display
XML	<code><text name="first_name">Tom</text></code>

html (isa dataitem)

Description	Holds HTML markup text.
Notes	The Controller may need to reformat the text for display on a given device. The contents must be quoted so as to be valid XML. Alternatively, a CDATA section could be used.
Attributes	
html	The html to display
XML	<code><html name="fancy_text">this is &lt;b&gt;bold&lt;/b&gt;</html></code>

number (isa dataitem)

Description	The base class for all data items that store numbers. It can hold either a single value or a value along with a set of historical values.
Notes	Can hold any number (double/long/fraction/positive/negative). The datasets (if any) embedded into the number will always have names of 'values', 'rates', and 'timestamps'. The "value" attribute will always store the current value in the array of historical values.
Attributes	
value	The last sampled value of the item
units	What the number represents (dollars, percent, seconds, etc.)
history	Number of values to keep (0 is assumed if absent). This can also be configured in the reaper.
values	<i>Dataset</i> of historical values; this is typically automatically created by the reaper
timestamps	<i>Dataset</i> of historical values; this is typically automatically created by the reaper
rate	Rate of change based on the <i>rate_type</i> ; the rate is typically created by the reaper
rates	<i>Dataset</i> of historical rates based on the <i>rate_type</i> ; the rates are typically created by the reaper; this is typically automatically created by the reaper
rate_type	Tells the reaper the type of rate to generate. This is described in detail in the "Reaper" portion of this document, and is typically set in the reaper configuration file.
XML	<u>Minimal:</u> <pre><number name='load_avg' timestamp='970169755' value='1.26'/></pre> <u>Everything:</u> <pre><number name='price' timestamp='970169746' value='1.49' units='\$' history='4' rate='-.5' rate_type='average:1'> <dataset name='values'>1.49, 2.01, 1.67, 1.80, 1.57</dataset> <dataset name='rates'>-.5, .34, -.13, .33, .57</dataset> <dataset name='timestamps'>970169750, 970169745, 970169740, 970169735, 970169730</dataset> </number></pre>

counter (isa number)

Description	A number that is non-negative and monotonically increasing.
Notes	Only increases until it is reset. If it ever decreases the reaper should throw an exception.
Attributes	

N/A	Same as <i>number</i> .
XML	<counter name='page_views' value='389222' timestamp='970169755'/>

gauge (isa number)

Description	A non-negative number that indicates some value that can fluctuate up and down (for example, system load or number of active users).
Notes	If the maximum or the minimum is not explicitly specified, then they are unbounded.
Attributes	
<i>maximum</i>	Known maximum value possible; reaper should throw an exception if it increases over this limit.
<i>minimum</i>	Known minimum value possible; reaper should throw an exception if it goes under this limit.
XML	<gauge name='active_users' timestamp='970169755' value='17' history='4' <dataset name='values'>17, 19, 21, 16, 20</dataset> <dataset name='timestamps'>970169755, 970169745, 970169735, 970169725, 970169715, </dataset> </gauge>

elapsed (isa number)

Description	Elapsed time in seconds, used to express relative time (for example, uptime or visit length).
Notes	Is converted by the Controller into an appropriate format (for example, HH:MM:SS)
Attributes	
N/A	
XML	<elapsed name='uptime' value='38921' timestamp='970169755'/>

chart (isa dataitem)

Description	A “chart” can store a set of axis/data labels plus multiple sets of values. Used instead of a <i>number</i> object when the axis is not time-based.
Notes	The number of items in the labels array, and in each values array must match the count attribute.
Attributes	
<i>count</i>	The number of values on the x-axis and the number of values in each <i>dataset</i>
<i>labels</i>	Array of labels for the x-axis
<i>dataset(s)</i>	<i>Dataset(s)</i> of y-values
XML	<chart name='fuel_prices' count='3'>

	<pre><labels>oil, gas, coal</labels> <dataset name='price'>35, 50, 15</dataset> <dataset name='tax'>20, 10, 12</dataset> </chart></pre>
--	---

labels (isa dataitem)

Description	Holds a list of labels (used in <i>chart</i>)
Notes	
Attributes	
labels	Array of textual or numeric labels
XML	See <i>chart</i> .

dataset (isa dataitem)

Description	Stores a list of values (see <i>chart</i> and <i>number</i> for example use)
Notes	
Attributes	
values	Array of y-values
XML	<pre><dataset name='price'>35, 50, 15</dataset></pre>

list (isa dataitem)

Description	Holds a list of other items.
Notes	The name of each item in the list must be unique.
Attributes	
N/A	
XML	<pre><list name='status'> <text name='summary'>All servers are operational</text> <list name='www1'> <number .../> <counter .../> <text name='status'> ... </text> </list> <list name='www2'> ... </list> </list></pre>

instruction (isa dataitem)

Description	Communicates an instruction back to the reaper.
Notes	
Attributes	
name	If the 'name' of an instruction data item is "noop", then the reaper will not record any information sent to it. Likewise, it will not update the item's history. This is useful for updating

	<p>data items that may not change frequently or have a period of inactivity (such as the stock market).</p> <p>If the 'name' of an <i>instruction</i> data item is "clear_history", the reaper will clear this data item's history datasets before inserting the current data. This is useful for "resetting" histories. For example, the stock market history could be cleared every morning when the market opens, so that a new graph is started for the day.</p>
XML	<pre><instruction name='noop'/> <instruction name='clear_history'/></pre>

Approved by the Board

Appendix C

NAME

Dashboard

DESCRIPTION

Driver class for Dashboard XML project.

SYNOPSIS

Creating (aka. freezing)

```
my $dashboard=new Dashboard();
```

Reading (aka. thawing)

```
my $dashboard=new Dashboard(xml => $xml_string);  
my $dashboard=new Dashboard(file => $file_name);
```

Accessing

```
$item=$dashboard->select('path/to/item/within/tree');
```

PREREQUISITES

XML::Parser

CONSTRUCTORS

The constructor can be called two ways: No arguments, which creates a brand-spankin' new data object, or with a chunk of XML or a file name, which will read and parse (``thaw") the XML, and construct a Dashboard data structure which can be queried, modified, and re-frozen as desired.

```
Dashboard::new(xml => 'some string of XML');
```

or

```
Dashboard::new(file => 'some_filename_containing_XML');
```

METHODS

select("selector/string/path")

This will search the tree for an item that can be reached by the specified list of ``selector" strings, which are separated by ``/" characters.

At each level in the tree, an item will be searched for that matches the given selector. For Lists, the Dataitem with the specified name will be returned. For other item types, an attribute of the given name will be searched for. This attribute may be a scalar value, or it might be an Array or a pointer to another Dataitem type.

Examples:

Given the following XML:

```
<dashboard>
  <list name="alpha">
    <gauge name='bravo' timestamp='970594169' value='111' />
    <list name='charlie'>
      <counter name='delta' historyLength='3' timestamp='970594170'
        value='20'>
        <dataset name='values'>20, 30, 40</dataset>
        <dataset name='timestamps'>970594163,970594164,
          970594165</dataset>
      </counter>
      <elapsed name='echo' value='2932' timestamp='970594168' />
    </list>
  </list>
  <number name='foxtrot' timestamp='970594179' value='333' />
</dashboard>
```

The following selectors will return the indicated objects (or die with an error as indicated):

alpha	List named "alpha"
bogus	die: 'bogus' not found
alpha/bravo	Gauge named "bravo"
alpha/bravo/value	Scalar:111
alpha/bravo/value/bogus/stuff	die: leaf at 'bogus'
alpha/bravo/bogus/stuff	die: 'bogus' not found
alpha/bravo/timestamp	Scalar:970594170
alpha/charlie	List named "charlie"
alpha/charlie/delta	Counter named "delta"
alpha/charlie/delta/value	Scalar:20
alpha/charlie/delta/historyLength	Scalar:3
alpha/charlie/delta/values	Dataset named "values"
alpha/charlie/delta/values/values	ARRAY[20,30,40]
alpha/charlie/delta/values/values/0	Scalar:20
alpha/charlie/delta/values/values/3	die: index '3' not found
alpha/charlie/delta/timestamps	Dataset named "timestamps"
alpha/charlie/delta/timestamps/values/1	Scalar:970594164
alpha/charlie/echo/value	Scalar:2932
foxtrot	Number named "foxtrot"
foxtrot/value	Scalar:333

Strange cases:

- To get the text of a text item use the selector ``.../itemname/text"
- To get the body of an html item use ``.../itemname/html"

- To get the URL of an image item use ``.../itemname/image"
- Charts:
 - To get the labels for a chart as an array use ``.../chartname/labels/labels".
 - To get a dataset for a chart as an array use ``.../chartname/datasets/0/values".
 - To get the name of a chart dataset use ``.../chartname/datasets/0/values".
 - To find out how many datasets there are use ``.../chartname/datasets/count".

select_scalar("selector/string/path")

Just like `select()`, except that it errors out if the item returned is not a scalar value.

xml_freeze()

Returns an XML string representation of the data structure.

xml_thaw(\$xml_string)

Returns a new Dashboard object created by parsing the specified XML stored in a string.
Preferred: use the 'xml' argument on the constructor.

add(LIST)

Adds a list of Dataitems to the current Dashboard object. Will die if an attempt to add an item named identially to an existing item in the Dashboard is made.

```
$dash=new Dashboard();  
$count=new Dashboard::Counter(value=>5);  
$dash->add($count);
```

or even:

```
$dash->add(new Dashboard::Counter(value=>5));
```

Appendix D

Controller Configuration Files

Navorg

Description	The navorg element defines the initial structure and contains attributes for defining the renderer and the name of the data configuration file to use.
Notes	
Attributes	
<i>name</i>	Informational name of this display configuration.
<i>aliases</i>	The path to the data configuration file. Optional, defaults to <i>conf/data.xml</i> .
<i>layout</i>	The name of the renderer that this user will use by default. The existing renderers are <i>framed</i> and <i>htmltable</i> . Defaults to <i>framed</i> .
<i>templatedir</i>	Template-override directory to use for this file. Can be overridden by <i>templatedir</i> specified in a <i>section</i> . Defaults to the username.
XML	<pre><navorg name="The Sporting News" aliases="tsn_data.xml" layout="framed"> ... </navorg></pre>

Section

Description	The <i>section</i> element defines the layout of a single display page. Each display page can have multiple cells (also called constructs or indicators) on it. <i>Sections</i> are linked together via the <i>onSelect</i> attributes of <i>constructs</i> (see below).
Notes	
Attributes	
<i>name</i>	The name of the section. The section <i>name</i> must be unique across all sections. Required. "start" is the <i>name</i> of the first page the user will view.
<i>template</i>	Defines the template directory to use for overriding the default template directory. Defaults to the <i>username</i> .
<i>templatedir</i>	Template-override directory to use for this file. Defaults to the <i>templatedir</i> specified in the <i>navorg</i> .
<i>rows</i>	The number of cells per column to create.
<i>cols</i>	The number of cells per rows to create.
<i>rowsize</i>	The height of each row in the display. Defaults to 150.
<i>colsize</i>	The width of each column in the display. Defaults to 200.

description	The description of the section. This is not available to the templates and is mainly for documentation purposes.
helptext	The help text for the section. If not specified, the cell will not contain a help link.
XML	<pre> <section name="start" description="Main Page" template="tsn"> <attr name="rows">4</attr> <attr name="cols">4</attr> ... </section> <section name="DJIA_chart" description="DJIA Chart"> ... </section> </pre>

Construct

Description	The construct element (also called a cell or an indicator) defines a business intelligence object. The construct itself defines a container type, a template file, the refresh rate, and the links for common navigation elements (drilldown, help, reports). The item elements enclosed by the construct define the mappings between values in the XML data files and variable names in the template file.
Notes	
Attributes	
name	The name of the construct. Must be unique across the enclosing <i>section</i> element.
position	The position of the cell in the matrix of cells, starting with 1 in the upper left corner, proceeding across the columns and then down to the next row. The <i>position</i> must be unique across the enclosing <i>section</i> . If the <i>position</i> is not unique, only the last <i>construct</i> with the given <i>position</i> will be displayed.
description	The description of the cell. This should be short, as it is typically displayed in the template as a title for the indicator.
template	The template name that this construct will use.
refresh	The refresh rate in seconds of this cell. This is typically calculated automatically when the dashboard data elements are loaded, as they themselves have the refresh rate of the data. The <i>refresh</i> is used to determine how frequently the indicator should try to refresh itself.
data_refresh	The frequency in seconds with which underlying data changes. This <i>data_refresh</i> number is typically displayed to the user on an indicator. If not specified, <i>refresh</i> is used. A typical use for <i>data_refresh</i> is for information that updates itself daily, but for which you want to check every hour to see if the information has changed. The <i>data_refresh</i> is to set it to 86400 (the

	number of seconds in a day), but the <i>refresh</i> is set to 3600 (the number of seconds in an hour). So, the user sees on the indicator that the data updates every 24 hours, but the actual indicator refreshes itself every hour, looking for potential new information.
<i>onSelect</i>	Defines the <i>section name</i> that the indicator's drilldown icon (magnifying glass) will be associated with. If not specified, the indicator will not have a drilldown icon.
<i>onSelect</i> <i>ScreenType</i>	Defines if the drilldown will be displayed in a separate popup window ("popup"), replaces the full screen ("full"), or replaces the current frame ("frame"). Defaults to "popup".
<i>precision</i>	Specifies a precision to limit all number elements to. The default is to leave numbers as-is. A precision of "1" will make numbers look like "2.5". This is handy for limiting the precision of averaged aggregate items.
XML	<pre><construct name="Control" description="Control" type="control" position="1"> </construct> <construct name="DJIAcurrent" description="DJIA" type="number" position="2" onselect="DJIA_graph" data_refresh="10"> <attr name="helptext"> &lt;U&gt;&lt;B&gt;Dow Jones Industrial Average&lt;/B&gt;&lt;/U&gt; &lt;P&gt; The large blue number is the current value of the DJIA. The arrow indicates the current direction of the market since the market last opened. The arrow will be green if the market is up for the day, and red if it's down. The small number under the arrow is the change since the market last opened. &lt;P&gt; The chart displays how the market has moved over the past 8 hours of activate trading. Every morning when trading begins, the chart begins anew. &lt;P&gt; This indicator is updated every 10 minutes. </attr> </construct></pre>

Item

Description	Item elements define the set of data to be made available for use in the display template. An <i>item</i> can only appear inside a set of <i>construct</i> tags.
Notes	

Attributes	
name	The name of the item. The <i>name</i> will in most cases also defines the name of the variable available in the template. An item <i>name</i> must be unique across the enclosing <i>construct</i> .
source	The alias name of the source object (from <i>data.xml</i>).
select	An additional path into the <i>source</i> item's object tree. Depending on the template variable that's being populated, the select will be either a leaf node (such as /sitestats/hits/value), or a non-leaf (such as /sitestats/hits/history). If a non-leaf is required for the template, the controller will do the work of gathering the required information out of the node.
format	<p>Sets a format for the controller to use for this item. Valid <i>formats</i> are:</p> <ul style="list-style-type: none"> • dollars (uses precision of 2 and pre-pends a "\$") • elapsed (converts to <i>DD</i> days, <i>HH:MM</i>) <p>The controller also automatically formats:</p> <ul style="list-style-type: none"> • If the <i>select</i> contains "timestamp", it will change the format to <i>MM/DD/YY HH:MM</i>. This is primarily used for auto-populating the x-axis on graphs. • If the item looks like a number (contains only digits and periods), it will: make a default precision of 2 and "comma-ize" the number. For example, 2003.456 will be changed to 2,003.46.
precision	Sets the numeric precision for the item.
type	<p>Defines a type of the particular item, which affects what is done with the data. Valid <i>types</i> are:</p> <ul style="list-style-type: none"> • graph • table • var • alert <p>Each of these <i>types</i> is explained in detail below.</p>
unit	Sets the unit string for the data item. The <i>unit</i> value is made available to the template in the <i>unit</i> variable.
prefix	Defines a prefix that is typically displayed before the item. The <i>prefix</i> value is made available to the template in the <i>prefix</i> variable.
graphtype	<p>The type of graph to display (only valid with a <i>type</i> of "graph"). Can be one of:</p> <ul style="list-style-type: none"> • bars • pie • lines <p>Defaults to "lines" if not specified.</p>
value	Sets the value of an <i>item</i> of <i>type</i> "var".

XML	<pre> <construct name="SysLoad" description="System Load" type="number" position="14"> <item name="alert" type="alert" source="ALERTS" select="SysLoad"/> <item name="graph" type="graph" source="systemload" select="load15"/> <item name="current" source="systemload" select="load15/value"/> <item name="history" source="systemload" select="load15/values/values"/> <item name="direction" source="systemload" select="load15/rate"/> <item name="timestamp" source="systemload" select="load15/timestamp"/> </construct> </pre>
------------	--

Item Types

var

Description	Defines a variable to set in the template.
Notes	
Attributes	
name	The name of the variable.
value	The value to set this variable to.
XML	<pre> <construct name="DJIA" description="DJIA Table" type="number" template="labels_only" position="1"> <item name="label1" type="var" value="Time"/> <item name="label2" type="var" value="Value"/> <item name="label3" type="var" value="Change"/> <item name="label4" type="var" value="% Change"/> </construct> </pre>

alert

Description	<p>If an alert is set for the <i>source/select</i> defined, then the <i>name</i> is used to populate a set of variables. If the <i>name</i> is set to "alert", then the following variables will be defined (based off of how the alert was set up in the alerts configuration file):</p> <ul style="list-style-type: none"> • alert_severity: The severity of the alert (0 through 3) • alert_msg: The alert message to display to the user • alert_notes: Notes (not currently used) • alert_level: The name of the alert level • alert_directional: Directional graphic suffix (e.g.: "red")
Notes	
Attributes	
name	The prefix of a template variable to populate.
source	The source file for the alerts.

select	The data item in the alerts file to look for.
XML	<pre><construct name="DJIA" description="DJIA" type="number" position="1"> <item name="alert" type="alert" source="ALERTS" select="DJIA"/> </construct></pre>

table

Description	Creates a table based on data elements supplied to it.
Notes	The current implementation can only grab data from one <i>source</i> . Labels should also be moved down into options, like “legend” is for the “graph” type.
Attributes	
name	The name of the table. The <i>name</i> is the name of the variable to be used in the template’s loop.
source	The source file for the table data.
option	<i>Options</i> are specified as elements of a <i>table</i> . Each <i>option</i> defines one data source for the table. The <i>option</i> contains a <i>name</i> attribute that defines the variable to populate in the template’s loop). The body of the <i>option</i> contains the select string into the <i>table</i> ’s data <i>source</i> .
XML	<pre><construct name="DJIA" description="DJIA Table" type="number" template="detail_table" position="1"> <item name="label1" type="var" value="Time"/> <item name="label2" type="var" value="Value"/> <item name="label3" type="var" value="Change"/> <item name="label4" type="var" value="% Change"/> <item name="table" type="table" source="DJIA"> <option name="data1">current/timestamps/values</option> <option name="data2">current/values/values</option> <option name="data3">difference/values/values</option> <option name="data4">percent/values/values</option> </item> </construct></pre>

graph

Description	Creates a graph based on data elements supplied to it.
Notes	The current implementation can only grab data from one <i>source</i> .
Attributes	
name	The name of the graph. The <i>name</i> is the name of the variable to be used in the template’s loop.
source	The source file for the graph data.
select	An additional path into the <i>source</i> item’s object tree.
option	<i>Options</i> specify a number of arguments to the graph: <ul style="list-style-type: none"> <i>axis_datanum</i>: If an option name has the format: x_datanum or y_datanum (where <i>num</i> is a data plot

	<p>number, starting with “1”), then the body of the <i>option</i> contains the select string into the <i>table</i>’s data <i>source</i>. If these types of <i>options</i> are not specified, then the graph’s <i>source</i> is used as a single element to graph. The x-axis is the <i>source</i>’s timestamps, and the y-axis is the <i>source</i>’s values.</p> <ul style="list-style-type: none">• <i>labelnum</i>: A label for the specified <i>num</i> data set. The default is to not have any labels on the graph.• <i>width</i>: The width of the graph. The default is 180.• <i>height</i>: The height of the graph. The default is 120.• <i>bgclr</i>: The background color of the graph. The default is yellow (255,255,204).
XML	<pre><construct name="HPLoad" description="HP Load Time" type="number" position="13"> <item name="graph" type="graph" source="servicemetrics"> <option name="width">500</option> <option name="height">200</option> <option name="bgclr">#ffffff</option> <option name="x_data1">site1/values/values</option> <option name="y_data1">site1/timestamps/values</option> <option name="legend1">Site 1</option> <option name="x_data2">site2/values/values</option> <option name="y_data2">site2/timestamps/values</option> <option name="legend2">Site 2</option> </item> </construct></pre>

Appendix E

Dashboard Controller Variables

Cause	An element's name is equal to "timestamp" or an element's <i>select</i> contains "timestamp".
Effect	Populates the following variables, with the element's <i>name</i> as the prefix: <ul style="list-style-type: none"> • <i>prefix_sec</i>: Seconds • <i>prefix_min</i>: Minutes • <i>prefix_hour</i>: Hour in 24-hour clock • <i>prefix_mday</i>: Day of the month • <i>prefix_mon</i>: Month number • <i>prefix_year</i>: 4-digit year • <i>prefix_mon_name</i>: 3-character month name • <i>prefix_time</i>: 12-hour time (e.g.: 1:23 PM) • <i>prefix_date</i>: Date in MM/DD/YY format (e.g.: 1/13/01)
Notes	A timestamp typically indicates a refresh time of the data object. It is supplied in the Unix time format (seconds).

Cause	The "description" attribute is populated (which is required for all elements anyway).
Effect	Populates a "description" variable.

Cause	A "graph" item exists in the construct.
Effect	<p>A "graph" variable is populated with the URL to generate the graph image.</p> <p>A "graph_map" variable is populated with the graph's imagedmap.</p> <p>A "width" variable is populated with the graph's width.</p> <p>A "height" variable is populated with the graph's height.</p>

Cause	The "refresh" or "data_refresh" attributes are populated.
Effect	<p>Populates a "refresh" variable that contains the refresh rate in the following format:</p> <ul style="list-style-type: none"> • 10 s (10 seconds) • 10 m (10 minutes) • 10 h (10 hours) • 10 d (10 days) <p>The controller figures out which format to use depending on how large the refresh number is.</p>
Notes	This currently doesn't deal with fractions (i.e.: 1.6 h is seen as 1h).

Cause	None
Effect	A “script” variable is populated with the URL that generated the current construct. This doesn’t have much use in the templates, but it’s used internally in the controller.

Cause	None
Effect	A “dashboardtop” variable is populated with the URL that generates the top-level of the dashboard. Useful for linking a “home” button.

Cause	onSelectScreen attribute is set in the <i>construct</i> .
Effect	One of “onSelectScreenFull”, “onSelectScreenFramed”, or “onSelectScreenPopup” variables is set to 1, depending on the value of <i>onSelectScreen</i> .

Cause	onSelect attribute is set in the <i>construct</i> .
Effect	“drilldown”, “drilldownframed”, “drilldowntable” variables are populated with the URL to the selected section (current format, framed format and table format, respectively). The URLs are auto-generated based on the onSelect attribute in this item’s <i>construct</i> .

Cause	BUG
Effect	An “onReport” attribute should be recognized inside the <i>construct</i> , but currently it’s not. Instead, we set it using a “var” type, which works out just fine.

Cause	unit attribute is populated within an <i>item</i> .
Effect	An <i>itemname</i> unit variable is populated with its value.
Notes	This could probably be moved into an <i>item</i> of type “var”.

Cause	prefix attribute is populated within an <i>item</i> .
Effect	An <i>itemname</i> prefix variable is populated with its value.
Notes	This could probably be moved into an <i>item</i> of type “var”.

Cause	An <i>item</i> named “direction” exists.
Effect	If the value of “direction” is: <ul style="list-style-type: none">• positive: Variable “directionup” is set to 1• zero: Variable “directionneutral” is set to 1• negative: Variable “directiondown” is set to 1
Notes	This is useful for determining if an up arrow or a down arrow should be displayed on the indicator. A direction item is typically the indicator’s rate: <item name="direction" source="sysload" select="cpu/rate"/>

Cause	An <i>item</i> named “history” exists.
Effect	The following variables are set: <ul style="list-style-type: none">• “high”: Highest value in the history• “low”: Lowest value in the history

Cause	An attribute named “helptext” exists.
Effect	A “helptext” variable is populated and a “helpurl” variable is populated with the URL to a help popup window

Cause	An “alert” type exists within the <i>construct</i> .
Effect	See the “alert” type above.

Appendix F

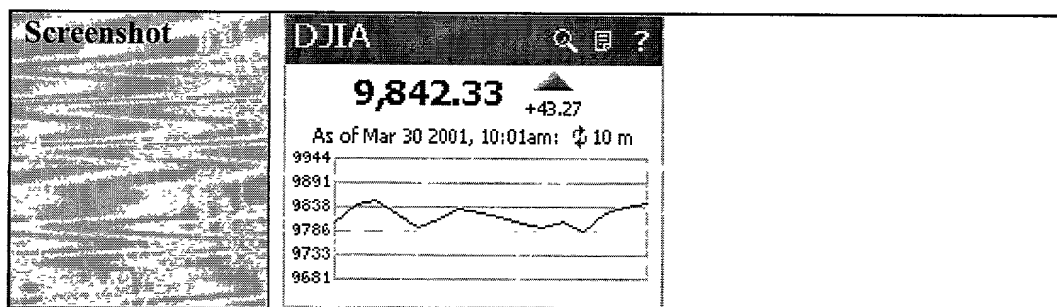
Dashboard Controller Templates

Number

Description	Displays a small, framed primary indicator with a graph beneath it of the indicator's history
Notes	This should really be renamed to "summary" instead of "number".
Primary Template	htmltable/number.tmpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/summary_titlebar.tmpl htmltable/summary_numbdisp.tmpl htmltable/summary_asofdate.tmpl htmltable/summary_graph.tmpl
Variables Used	
Titlebar	
Description	Used as the title for the indicator
<i>drilldown table</i>	If populated, a detail icon will be created with a link to the URL contained in <i>drilldown table</i> . Note that this variable is automatically created by the controller if <i>onSelect</i> is populated.
<i>drilldown framed</i>	If populated, and <i>onselectscreenframe</i> is also populated, a detail icon will be created with a link to the URL contained in <i>drilldown framed</i> . Note that this variable is automatically created by the controller if <i>onSelect</i> is populated.
<i>onSelect ScreenFull</i>	If populated, and <i>drilldown framed</i> is populated, a detail icon will be created with a link to the URL contained in <i>drilldown framed</i> with target="_top". Note that this variable is automatically created by the controller if <i>onSelectScreen</i> is populated.
<i>onSelect ScreenFrame</i>	If populated, and <i>drilldown framed</i> is populated, a detail icon will be created with a link to the URL contained in <i>drilldown framed</i> , to appear in the existing frame. Note that this variable is automatically created by the controller if <i>onSelectScreen</i> is populated.
<i>reporturl</i>	If populated, a report icon link will be created to the URL contained in <i>reporturl</i> .
<i>onReport ScreenFull</i>	If <i>reporturl</i> is populated, adds target="_top" to the link. By default, a <i>reporturl</i> is displayed in a popup window.
<i>onReport ScreenFrame</i>	If <i>reporturl</i> is populated the <i>reporturl</i> will be displayed in the current frame. By default, a <i>reporturl</i> is displayed in a popup window.

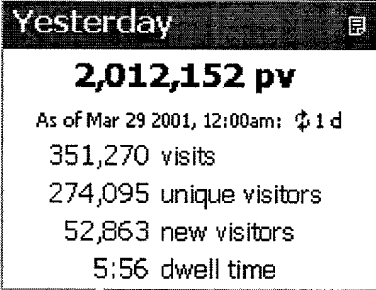
<i>helpurl</i>	If populated, a help icon link will be created to the URL contained in <i>helpurl</i> . Note that this variable is created automatically by the controller if <i>helptext</i> is populated.
Numbdisp	
current	The primary indicator number, displayed in bold, as populated by an <i>item</i> named “current”.
<i>current_unit</i>	If a <i>unit</i> is specified within the “current” <i>item</i> , it will be placed directly after the primary indicator (e.g.: 123 page views).
<i>direction_up</i>	One of <i>directionup</i> , <i>directiondown</i> , or <i>directionneutral</i> is set by the controller depending on the value specified by the <i>difference</i> item.
<i>direction_down</i>	See <i>directionup</i> .
<i>direction_neutral</i>	See <i>directionup</i> .
<i>alert_directional</i>	<i>alert_directional</i> is set if an alert is found for this indicator. The directional is used as a suffix for the up/down image. For example, if <i>alert_directional</i> is set to “red”, then “upred.gif” or “downred.gif” will be leaded as the up/down image.
<i>alert_severity</i>	<i>alert_severity</i> is set if an alert is found for this indicator. The severity is used as a suffix for the alert icon. For example, if <i>alert_severity</i> is set to “2”, then “alert1.gif” is used for the alert image. If <i>alert_severity</i> is 0, then no alert image is displayed. <i>alert_severity</i> is also displayed when the user mouses over the alert icon.
<i>alert_msg</i>	<i>alert_msg</i> is set if an alert is found for this indicator. The msg is displayed along with the alert’s severity when the user mouses over the alert icon.
<i>direction</i>	The current “direction” of the indictor, as populated by an <i>item</i> named “direction”. This is typically the current rate of change.
<i>direction_unit</i>	If a <i>unit</i> is specified within the “direction” <i>item</i> , it will be placed directly after the direction (e.g.: -3.4%).
Asofdate	
timestamp_mon_name	If an <i>item</i> named “timestamp” is populated, the controller will automatically populate these <i>timestamp_*</i> variables. These variables are used to tell the end-user when the last time the data was updated.
timestamp_mday	See <i>timestamp_mon_name</i> .
timestamp_year	See <i>timestamp_mon_name</i> .
timestamp_	See <i>timestamp_mon_name</i> .

time	
data_refresh	If the <i>construct</i> contains a <i>data_refresh</i> attribute, or the <i>construct</i> contains a <i>screen_refresh</i> item, then this variable will automatically be populated by the controller. This variable is used to tell the user how frequently the data is updated.
Graph	
graph	If a “graph” <i>item</i> is contained in the <i>construct</i> , then the controller will automatically populate this variable. The variable contains a link to the URL that will display the graph.
drilldown table	If this variable is populated (see the “Titlebar” section), then the graph is also linked to this URL.
description	The <i>description</i> is used to populate the ALT tag of the graph.
XML	<pre> <construct name="DJIAcurrent" description="DJIA" type="number" position="2" onselect="DJIA_graph"> <item name="reporturl" type="var" value="http://quote.yahoo.com/q?s=^dji&amp;d=b"/> <item name="alert" type="alert" source="ALERTS" select="DJIA"/> <item name="alert_per" type="alert" source="ALERTS" select="DJIA_percent"/> <item name="graph" type="graph" source="DJIA" select="current"/> <item name="current" source="DJIA" select="current/value"/> <item name="history" source="DJIA" select="current/values/values"/> <item name="direction" source="DJIA" select="difference/value"/> <item name="percent" source="DJIA" select="percent/value"/> <item name="timestamp" source="DJIA" select="current/timestamp"/> <item name="screen_refresh" source="DJIA" select="current/refresh"/> <attr name="helptext"> ... </attr> </construct> </pre>



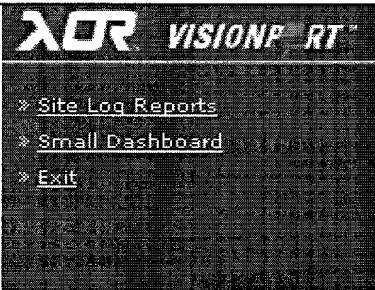
Summary_stats

Description	Displays a small, framed primary indicator with a set of supporting statistics beneath it.
Primary Template	htmltable/summary_stats.tpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/summary_titlebar.tpl htmltable/summary_statupper.tpl htmltable/summary_asofdate.tpl htmltable/summary_statslower.tpl
Variables Used	
Titlebar	See the “Titlebar” section under the Number template.
statsupper	
val1	The primary indicator number, displayed in bold, as populated by an <i>item</i> named “val1”.
val1_unit	If a <i>unit</i> is specified within the “val1” <i>item</i> , it will be placed directly after the primary indicator (e.g.: 123 page views).
val1_prefix	If a <i>prefix</i> is specified within the “val1” <i>item</i> , it will be placed directly before the primary indicator (e.g.: Page Views: 123).
val1rate	The rate of change for the primary indicator, as populated by an <i>item</i> named “val1rate”.
val1rate_unit	If a <i>unit</i> is specified within the “val1rate” <i>item</i> , it will be placed directly after the rate (e.g.: -1.23%).
Asofdate	See the “Asofdate” section under the Number template.
statslower	
valx	A statistic, as populated by an <i>item</i> named “valx”, where x is a number from 2 to 5.
valx_unit	If a <i>unit</i> is specified within the “valx” <i>item</i> , it will be placed directly after the primary indicator (e.g.: 123 page views).
valx_prefix	If a <i>prefix</i> is specified within the “valx” <i>item</i> , it will be placed directly before the primary indicator (e.g.: Page Views: 123).
valxrate	The rate of change for the primary indicator, as populated by an <i>item</i> named “valxrate”.
valxrate_unit	If a <i>unit</i> is specified within the “valxrate” <i>item</i> , it will be placed directly after the rate (e.g.: -1.23%).

XML	<pre> <construct name="SiteStats_yesterday" description="Yesterday" type="summary_stats" data_refresh="86400" position="9"> <item name="reporturl" type="var" value="/cgi- bin/webtrends_redirect/5/1"/> <item name="val1" source="historical_stats" select="all/yesterday/pageviews/value" unit="pv"/> <item name="timestamp" source="historical_stats" select="all/yesterday/pageviews/timestamp"/> <item name="screen_refresh" source="historical_stats" select="all/yesterday/pageviews/refresh"/> <item name="val2" source="historical_stats" select="all/yesterday/visits/value" unit="visits"/> <item name="val3" source="historical_stats" select="all/yesterday/unique_visitors/value" unit="unique visitors"/> <item name="val4" source="historical_stats" select="all/yesterday/firsttime_visits/value" unit="new visitors"/> <item name="val5" source="historical_stats" select="all/yesterday/dwell/value" unit="dwell time" format="elapsed"/> </construct> </pre>
Screenshot	 <p>Yesterday</p> <p>2,012,152 pv</p> <p>As of Mar 29 2001, 12:00am: 1 d</p> <p>351,270 visits</p> <p>274,095 unique visitors</p> <p>52,863 new visitors</p> <p>5:56 dwell time</p>

Control, Control_subpage

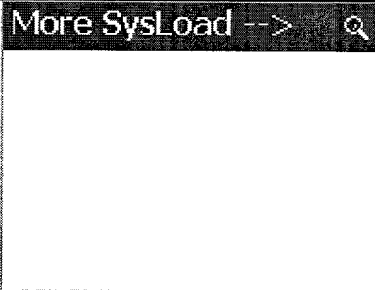
Description	Displays a small, framed "Control Panel". The only difference between Control and Control_subpage is the inclusion of a "back" button on the subpage version.
Primary Template	htmltable/control.tmpl htmltable/control_subpage.tmpl
Templates Included	htmltable/visionport.css htmltable/visionport.js
Variables Used	
dashboard top	Link to the "top level" of the dashboard. This variable is automatically populated by the controller.

weblog	If populated, the <i>weblog</i> variable will link the user to a 3 rd party product for viewing their web logs.
weblog_desc	Description shown to the user if <i>weblog</i> is populated.
weblog2	Second <i>weblog</i> , typically populated if the user has two weblog profiles to view.
weblog_desc2	Description for the second <i>weblog</i> .
XML	<pre><construct name="Control" description="Control" type="control" position="1"> <item name="weblog" type="var" value="http://visionport.xor.com/webtrends/demo/" /> <item name="weblog_desc" type="var" value="Site Log Reports" /> </construct></pre>
Screenshot	

Clientlogo

Description	Displays a small, framed logo.
Primary Template	htmltable/clientlogo.tmpl
Variables Used	
image	URL to the image to display.
width	Width of the image.
height	Height of the image.
alt	ALT text for the image.
url	Link to the customer's web site
XML	<pre><construct name="Logo" description="Logo" type="clientlogo" position="16"> <item name="image" type="var" value="http://www.client.com/images/logo.gif" /> <item name="width" type="var" value="100" /> <item name="height" type="var" value="50" /> <item name="alt" type="var" value="Client Site" /> <item name="url" type="var" value="http://www.client.com" /> </construct></pre>

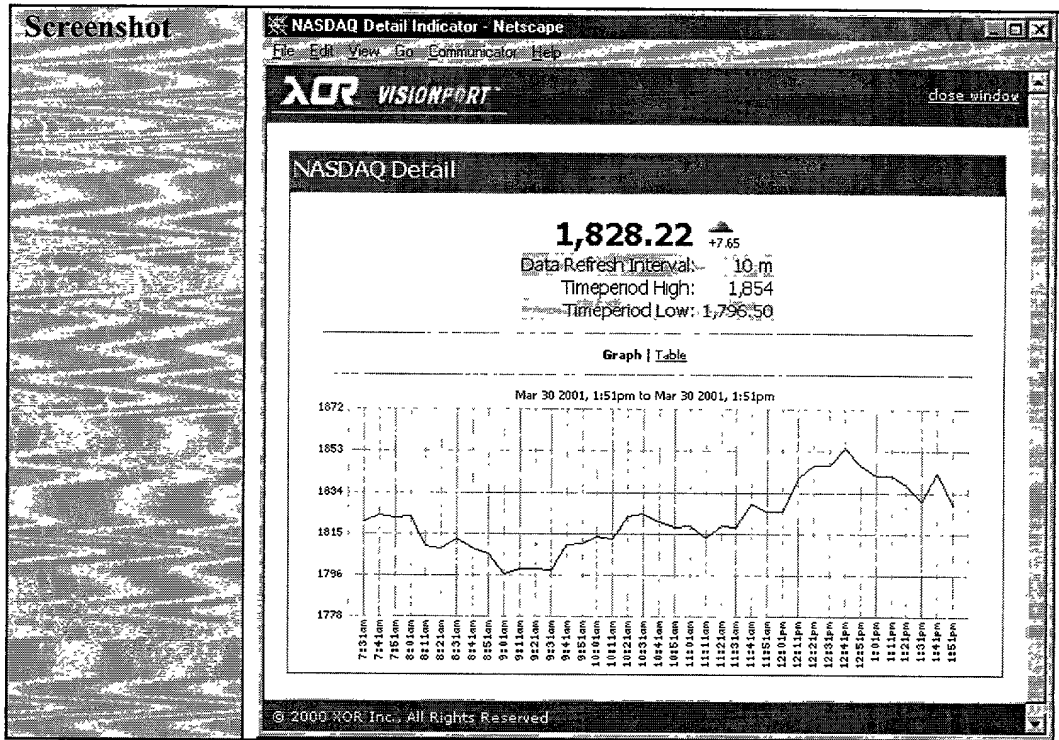
Blank

Description	Displays only a title bar. This is used for displaying an indicator that only has links to a detail and/or report icon. Useful for displaying a “More Indicators” cell.
Primary Template	htmltable/blank.tmpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/summary_titlebar.tmpl
Variables Used	
Titlebar	See the “Titlebar” section under the Number template.
XML	<construct name="next" description="More SysLoad -->" type="blank" position="2" onselectscreen="full" onselect="SysLoad_graph2"> </construct>
Screenshot	

Detail_graph

Description	Displays a larger version of a “Number” template. The graph is significantly larger and contains labels on the x- and y-axis. High and lows from the graph are also displayed.
Primary Template	htmltable/detail_graph.tmpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/detail_titlebar.tmpl htmltable/detail_numbdisp.tmpl htmltable/detail_graph_toggle.tmpl htmltable/detail_graph_body.tmpl htmltable/detail_footer.tmpl
Titlebar	
description	Used for the title of the page.
Numbdisp	See the “Numbdisp” section under the Number template. The following variables are also used:
data_refresh	If the <i>construct</i> contains a <i>data_refresh</i> attribute, or the <i>construct</i> contains a <i>screen_refresh</i> item, then this variable will automatically be populated by the controller. This variable is used to tell the user how frequently the data is

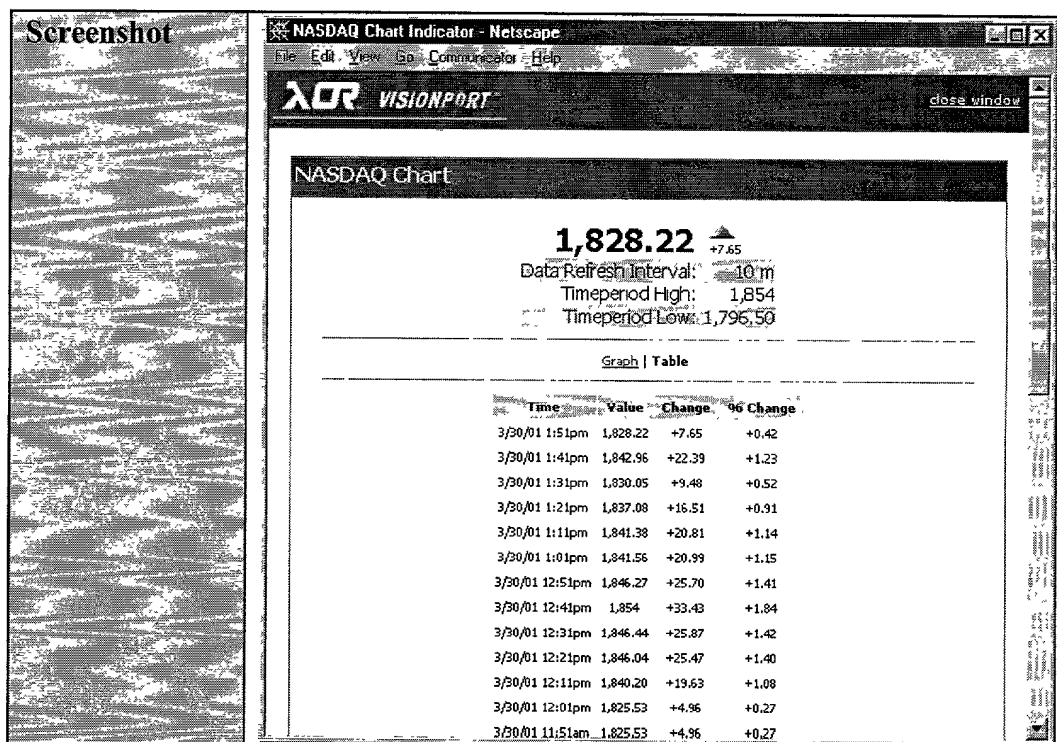
	updated.
<i>high</i>	The <i>high</i> is automatically populated if the <i>construct</i> has a "history" <i>item</i> defined.
<i>low</i>	The <i>low</i> is automatically populated if the <i>construct</i> has a "history" <i>item</i> defined.
Graph_toggle	
Drilldown table	Used to toggle between a graph and a table. The <i>onSelect</i> should be set to the table version of this graph.
Graph_body	See the "graph" section of the Number template. Also uses the timestamp variables <i>starttime</i> and <i>endtime</i> .
Footer	No variables; just an HTML footer with XOR's copyright.
XML	<pre> <section name="NASDAQ_graph" description="NASDAQ Detail" template="tsn"> <construct name="NASDAQ" description="NASDAQ Detail" type="number" position="1" onselect="NASDAQ_chart" template="detail_graph"> <item name="graph" type="graph" source="NASDAQ" select="current"> <option name="width">500</option> <option name="height">200</option> <option name="bgclr">#ffffff</option> </item> <item name="current" source="NASDAQ" select="current/value"/> <item name="history" source="NASDAQ" select="current/values/values"/> <item name="direction" source="NASDAQ" select="difference/value"/> <item name="percent" source="NASDAQ" select="percent/value"/> <item name="starttime" source="NASDAQ" select="current/timestamps/values"/> <item name="endtime" source="NASDAQ" select="current/timestamp"/> <item name="timestamp" source="NASDAQ" select="current/timestamp"/> <item name="screen_refresh" source="NASDAQ" select="current/refresh"/> </construct> </section> </pre>



Detail table

Description	Displays a table of the raw data for a set of information. This is typically linked with a detail_graph template.
Primary Template	htmltable/detail_table.tmpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/detail_titlebar.tmpl htmltable/detail_numbdisp.tmpl htmltable/detail_table_toggle.tmpl htmltable/detail_table_body.tmpl htmltable/detail_footer.tmpl
Variables Used	
Titlebar	See the "Titlebar" section under the Detail_graph template.
Numbdisp	See the "Numbdisp" section under the Detail_graph template.
Table_toggle	See the "Graph_toggle" section under the Detail_graph template.
Table_body	
labelx	Labels for the chart, where x is 1 through 5
datax	Data items that are looped over to populate the chart. x is a number 1 through 5.
Footer	No variables; just an HTML footer with XOR's copyright.
XML	<section name="NASDAQ_chart" description="NASDAQ Chart" template="tsn">

<construct name="NASDAQ" description="NASDAQ Chart" type="number" template="detail_table" position="1" onselect="NASDAQ_graph">
 <item name="label1" type="var" value="Time"/>
 <item name="label2" type="var" value="Value"/>
 <item name="label3" type="var" value="Change"/>
 <item name="label4" type="var" value="% Change"/>
 <item name="table" type="table" source="NASDAQ">
 <option
name="data1">current/timestamps/values</option>
 <option name="data2">current/values/values</option>
 <option name="data3">difference/values/values</option>
 <option name="data4">percent/values/values</option>
 </item>
 <item name="current" source="NASDAQ"
select="current/value"/>
 <item name="history" source="NASDAQ"
select="current/values/values"/>
 <item name="direction" source="NASDAQ"
select="difference/value"/>
 <item name="percent" source="NASDAQ"
select="percent/value"/>
 <item name="starttime" source="NASDAQ"
select="current/timestamps/values"/>
 <item name="endtime" source="NASDAQ"
select="current/timestamp"/>
 <item name="timestamp" source="NASDAQ"
select="current/timestamp"/>
 <item name="screen_refresh" source="NASDAQ"
select="current/refresh"/>
</construct>
</section>



Detail_stats

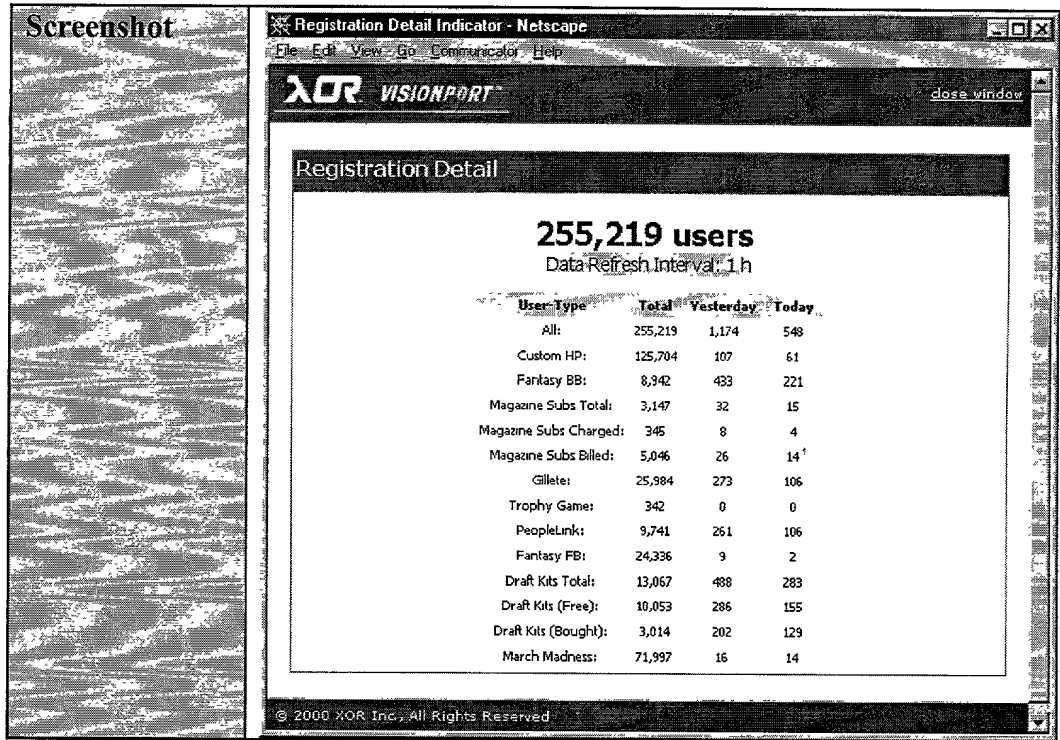
Description	Displays a larger version of a "Number" template. The graph is significantly larger and contains labels on the x- and y-axis. High and lows from the graph are also displayed.
Primary Template	htmltable/detail_stats.tpl
Templates Included	htmltable/visionport.css htmltable/visionport.js htmltable/detail_titlebar.tpl htmltable/detail_numbdisp.tpl htmltable/detail_stats_body.tpl htmltable/detail_footer.tpl
Variables Used	
Titlebar	See the "Titlebar" section under the Detail_graph template.
Numbdisp	See the "Numbdisp" section under the Detail_graph template.
Stats_body	
labelx	Labels for the chart, where x is 1 through 5
Data_x_y	Data items that are used to populate the table of statistics. x and y range from 1 to 5. There's probably a much better way to do this than the way it's currently coded...
Footer	No variables; just an HTML footer with XOR's copyright.
XML	<construct name="RegUsersDetail" description="Registration Detail" type="number" template="detail_stats" position="1" data_refresh="3600">

<item name="label1" type="var" value="User-Type"/>
<item name="label2" type="var" value="Total"/>
<item name="label3" type="var" value="Yesterday"/>
<item name="label4" type="var" value="Today"/>

<item name="data1_1" type="var" value="All:"/>
<item name="data1_2" source="reg_history"
select="appusers/all/All/value"/>
<item name="data1_3" source="reg_history"
select="appusers/yesterday/All/value"/>
<item name="data1_4" source="reg_today"
select="appusers/today/All/value"/>

<item name="data2_1" type="var" value="Custom HP:"/>
<item name="data2_2" source="reg_history"
select="appusers/all/16/value"/>
<item name="data2_3" source="reg_history"
select="appusers/yesterday/16/value"/>
<item name="data2_4" source="reg_today"
select="appusers/today/16/value"/>

<item name="data3_1" type="var" value="Fantasy BB:"/>
<item name="data3_2" source="reg_history"
select="appusers/all/23/value"/>
<item name="data3_3" source="reg_history"
select="appusers/yesterday/23/value"/>
<item name="data3_4" source="reg_today"
select="appusers/today/23/value"/>
...
</construct>



Show_help

Description	Template used for the popup help window.
Primary Template	htmltable/blank.tmpl
Templates Included	htmltable/visionport.css htmltable/detail_titlebar.tmpl htmltable/detail_footer.tmpl
Variables Used	
helptext	Help text, as defined a <i>construct's helptext</i> attribute.
Titlebar	See the "Titlebar" section under the Detail Graph template.
Footer	See the "Footer" section under the Detail Graph template.
XML	N/A
Screenshot	N/A

Small Dashboard

Description	A collection of the summary_* templates that are used to display a smaller version of the dashboard.
Primary Template	
Templates Included	htmltable/small/*.tmpl
Variables Used	See the normal templates for the variables. There is no additional work required in the controller configuration file to

	use the small templates. The dashboard has a “switch”, controlled by the <i>subtemplate</i> QUERY_STRING variable on the URL. When the controller sees this set, it first looks in the named <i>subtemplate</i> directory for the template first, and then defaults to the normal template directory.
XML	N/A
Screenshot	N/A

Appendix G

Dashboard Indicator Details

Indicator	Data Details
DJIA, NASDAQ, S&P 500	The Dow Jones Industrial Average, NASDAQ index and Standard and Poors 500 index. This information is gathered from Yahoo Financial by "screen scraping" the data from their HTML pages. When XOR has a client that wants this information a stock subscription service will be used.
Today's Stats, Today's Non- web	This information is gathered every 10 minutes from the client's different web servers. A process runs on each server and constantly collects information about visits, hits, page views and content groups being served. Note that this currently only runs on Unix systems. This indicator also displays the number of new registrations received for the current day and the number of visits from users that have registered previously on the site. Selecting the "detail" icon displays a breakdown of statistics for each server.
Today's Ads Served	The client that this data is taken from displays banner ads on their site using a 3 rd party ad-serving program called RealMedia OpenAdstream. XOR developed a Dashboard agent that parses the log files of the ad-server and reports on each ad "position" being served. Selecting the "detail" icon displays a breakdown of all the different ad positions.
Today's Revenue	The revenue number is a combination of dollars per visitor (this is how the client estimates their advertisement and sponsorship revenue) and actual subscription services sold on the site. Subscription information is extracted from the client's Informix database.
Yesterday, 8 Days ago, 7 Day Average, 28 Day Average, Visits Per Day	These statistics are gathered either from XOR's proprietary log analysis tool (all clients hosted by XOR receive a monthly report generated by this tool) or from another 3 rd party log analysis package, such as CommerceTrends. In this case, CommerceTrends was used. Selecting the "report" icon brings the user into the CommerceTrends report for the selected date.

HP Load Time	XOR has a partnership with a Service Metrics, a company that measures download times for a given URL from multiple points around the Internet. This gives the client a good idea about an average user's "download time experience". The graph displays two different views of this client's site: downloading their homepage with ads and without ads. This was done to help pinpoint problems with a 3 rd party ad service (24x7 Media) that caused the client's homepage to consistently load at least two seconds slower when using 24x7 Media's service. The third line on the graph is an "Internet Index" that Service Metrics provides. This is the average download time of 100 large Internet sites. There's another index, the "eCommerce Index" that can also be displayed with this graph. Service Metrics is a standard option of the VisionPort offering to which a client can subscribe.
Average System Load	This measures the average system load across the client's four main servers. Typically, the higher the load, the busier the machine. This information is gathered on-demand from each machine's operating system statistics. Selecting the "detail" icon displays system details about each individual machine.
Registered Users	This particular client has a registration system on their site. The indicator displays the number of users that have registered for various services. The main Dashboard screen displays the services that the client is most interested in watching. This indicator is changed at the client's request to display other registration types. Selecting the "detail" icon displays a breakdown of all types of registered users.